



TUGAS AKHIR - TE 141599

**RANCANG BANGUN SEPATU DENGAN PENGIKAT TALI
OTOMATIS**

HIBBAN KALDERA
NRP 0711124000059

Dosen Pembimbing
Dr. Ir. Djoko Purwanto, M.Eng.
Astria Nur Irfansyah, ST., M.Eng.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

DEVELOPMENT OF AUTOMATIC LACING SHOES

HIBBAN KALDERA
NRP 07111240000059

Supervisor
Dr. Ir. Djoko Purwanto, M.Eng.
Astria Nur Irfansyah, ST., M.Eng.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “Rancang Bangun Sepatu dengan pengikat tali otomatis” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 17 Januari 2018

Hibban Kaldera
NRP. 0711124000059

**RANCANG BANGUN SEPATU DENGAN PENGIKAT TALI
OTOMATIS**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Elektronika
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Dosen Pembimbing I,

Dosen Pembimbing II,



Dr. Ir. Djoko Purwanto, M.Eng.
Nip: 196512111990021002



Astria Nur Irfansyah, ST., M.Eng.
Nip: 198103252010121002



PERPUSTAKAAN	
ITS	
Tgl. Terima	01/02/2018
Terima Dari	H
No. Agenda	—

Rancang Bangun Sepatu dengan Pengikat Tali Otomatis

Hibban Kaldera
07111240000059

DosenPembimbing I : Dr. Ir. Djoko Purwanto, M.Eng.
DosenPembimbing II : Astria Nur Irfansyah, ST., M.Eng.

Abstrak:

Banyak film fiksi ilmiah yang menjadi pemicu penerapan teknologi di masa depan. Salah satunya yang sudah diterapkan sepatu berteknologi canggih yang dapat mengikat tali sendiri. Pada tugas akhir ini penulis terinspirasi dari film di atas untuk membuat suatu produk purwarupa sepatu dengan pengikat otomatis yang menggunakan motor DC sebagai penggulung talinya yang dapat membantu penyandang tremor dalam menggunakan sepatu. Motor DC akan menggulung dan mengendurkan tali dengan tombol yang tersedia dan juga dapat menggulung otomatis ke kekencangan tertentu. Terdapat tiga buah tombol berfungsi sebagai pengatur kekencangan, tombol pengencang yang berfungsi mengencangkan tali dan jika ditekan dua kali secara cepat akan mengencangkan ke kekencangan preferensi, tombol pengendur yang berfungsi untuk mengendurkan dan apabila ditekan dua kali secara cepat akan mengendurkan tali ke pengenduran maksimal, dan tombol preferensi yang akan menyimpan nilai kekencangan yang sesuai dengan pengguna. Terdapat dua sensor berat yang dipasang pada sol sepatu yang akan memberi masukan nilai ke prosesor mikrokontroler yang apabila mencapai nilai tertentu akan memicu pengikatan otomatis menuju kekencangan preferensi. Purwarupa ini menggunakan dua buah mikrokontroler pro micro yang salah satunya digunakan sebagai kontroler utama dan yang lainnya sebagai pemroses sensor, dua buah sensor berat dengan berat maksimum 50 KG, ADC HX711 untuk sensor berat, dua buah relay 1-channel, dan sebuah motor DC JA12-N20. Sistem dapat bekerja dengan baik tetapi motor tidak dapat menggulung tali sepenuhnya karena kurangnya torsi motor.

Kata kunci: Sensor berat, pengikat otomatis, Motor DC

Halaman ini sengaja dikosongkan

Development of Automatic Lacing Shoes

Hibban Kaldera
07111240000059

Supervisor I : Dr. Ir. Djoko Purwanto, M.Eng.
Supervisor II : Astria Nur Irfansyah, ST., M.Eng.

Abstract:

Many science fiction movies becomes the triggers of future technologies. One that have been implemented is shoes with sophisticated technologies which can do self-lacing to help people with tremor disorder to lace their shoes. In this final project writer build a prototype of self-lacing shoes mith DC motors as the roller inspired by movie above. DC Motors will tighten and loosen the straps by using switches and also able to automatically tightens the straps to a particular fit if a stomp is applied. There are three switches that works as the fit controller, the first one is the tightener switch which tightens the strap and if pressed twice consecutively will tighten to the preference tightness, the second is loosener switch which will loosen the strap and if pressed twice consecutively will loose the straps into the maximum loose, the third one is preference switch which will save the current tightness as preference tightness. There are two load sensors attached inside the sole which will give input value to the microcontroller processor which if specific value is achieved will trigger automatic tightening into optimal fit. This prototype uses two pro micro microcontrollers, one used as the main controller and the other as sensors processor, two load sensors with maximum load 50 KG, HX711 ADC for the load sensors, two 1-channel 5v relays, and a JA12-N20 DC Motors. The system works well but the motor can't fully lace because of the lack of torque.

Keyword: Load Sensors, Automatic Lacing, DC Motors

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Alhamdulillahirobbilalamin, segala puji syukur kepada Allah SWT, atas segala nikmat, berkah, dan hidayah-Nya yang tak terkira kepada penulis, hingga penulis mampu menyelesaikan Tugas Akhir dengan judul :

Rancang Bangun Sepatu dengan Pengikat Tali Otomatis

Tujuan utama tugas akhir ini adalah sebagai salah satu persyaratan untuk menyelesaikan jenjang pendidikan pada Bidang Studi Elektronika Teknik Elektro Institut Teknologi Sepuluh Nopember.

Atas selesainya penyusunan tugas akhir ini, penulis ingin mengucapkan terima kasih kepada :

1. Dr. Ir. Djoko Purwanto, M.Eng. dan Astria Nur Irfansyah, ST., M.Eng. selaku dosen pembimbing Tugas Akhir yang telah memberi bimbingan, masukan, nasehat, dan kemudahan dalam penyelesaian Tugas Akhir ini.
2. Dr. Tri Arief Sardjono, ST., MT., Ir. Tasripan, MT., dan Dr. Ronny Mardiyanto, ST., MT. selaku dosen penguji yang telah megoreksi Tugas Akhir ini.
3. Dr. Eng. Ardyono Priyadi, ST., M.Eng. selaku ketua Departemen Teknik Elektro ITS
4. Teman-teman IST Surabaya yang memberi masukan ide dan selalu mendukung proses pengerjaan tugas akhir ini serta Ach. Ibtihal Madarik Al-Fikry yang membantu mendesain konsep tugas akhir ini.
5. Seluruh keluarga besar terutama kedua orang tua yang selalu memberi dorongan dan untuk menyelesaikan tugas akhir ini. Juga Fedric “Mboot” Fernando yang selalu membantu dikala kesulitan.
6. Teman-teman dari E52 yang tidak dapat disebutkan satu persatu serta semua pihak yang telah membantu penulis dalam menyelesaikan

Penulis berharap para pembaca Tugas Akhir ini bersedia memberikan kritik, saran, dan masukan agar selanjutnya menambah manfaat Tugas Akhir. Semoga laporan Tugas Akhir ini dapat bermanfaat dan bisa dijadikan referensi bagi Tugas Akhir selanjutnya.

Halaman ini sengaja dikosongkan

DAFTAR ISI

HALAMAN JUDUL	
LEMBAR PERNYATAAN KEASLIAN	
LEMBAR PENGESAHAN	
ABSTRAK.....	i
ABSTRACT.....	iii
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
TABLE OF CONTENT.....	ix
DAFTAR GAMBAR.....	xi
BAB I PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	1
1.3. Tujuan Penelitian.....	2
1.4. Metodologi Penelitian.....	2
1.5. Sistematika Penulisan.....	3
1.6. Relevansi.....	4
BAB II DASAR TEORI	
2.1. Sensor Berat.....	5
2.1.1. Prinsip Kerja Sensor Berat.....	5
2.1.2. Menentukan Perilaku dari Sensor.....	6
2.1.3. Jembatan Wheatstone.....	7
2.2. 24-Bit ADC HX711.....	8
2.3 Motor DC JA12 N20.....	10
2.3.1 Analisa Motor DC.....	11
2.4. Mikrokontroler Pro Micro.....	13
2.5. Modul Relay.....	14

BAB III PERANCANGAN SISTEM

3.1	Diagram Blok Sistem.....	17
3.2.	Perancangan Sistem tombol.....	18
3.2.1	Perancangan Perangkat Keras.....	18
3.2.2	Perancangan Perilaku Tombol Pengencang.....	20
3.2.3	Perancangan Perilaku Tombol Pengendur.....	21
3.2.4	Perancangan Perilaku Tombol Preferensi.....	22
3.2.5	Perancangan Perilaku Tombol Reset.....	23
3.3.	Perancangan Sistem Sensor	23
3.4.	Perancangan Sistem Motor.....	24

BAB IV PENGUKURAN DAN ANALISIS SISTEM

4.1.	Pemeriksaan Kondisi Sistem Tombol.....	27
4.2.	Pemeriksaan Sistem Sensor.....	29

BAB V PENUTUP

5.1.	Kesimpulan.....	33
5.2.	Saran.....	33

DAFTAR PUSTAKA.....	37
---------------------	----

LAMPIRAN.....	49
---------------	----

BIOGRAFI PENULIS.....	55
-----------------------	----

TABLE OF CONTENT

COVER	
AUTHENTICITY	
APPROVAL SHEET	
ABSTRACT.....	i
ABSTRACT.....	iii
PREFACE.....	v
TABLE OF CONTENT.....	vii
TABLE OF CONTENT.....	ix
ILLUSTRATION.....	xi
CHAPTER I INTRODUCTION	
1.1. Background.....	1
1.2. Problem Formulation.....	1
1.3. Research Objective.....	2
1.4. Methodology.....	2
1.5. Writing Systematic.....	3
1.6. Relevance.....	4
CHAPTER II LITERATRE OBSERVATION	
2.1. Load Sensors.....	5
2.1.1. Load Sensors Working Principle.....	5
2.1.2. Deciding Sensors Behaviour.....	6
2.1.3. Wheatstone Bridge.....	7
2.2. 24-Bit HX711 ADC.....	8
2.3 JA12 N20 DC Motors.....	10
2.3.1 DC Motors Analysis.....	11
2.4. Pro Micro Microcontroller.....	13
2.5. Relay Module.....	14

CHAPTER III SYSTEM DESIGN

3.1 System Block Diagram.....17

3.2. Design of Switch System.....18

3.2.1 Design of Hardware.....18

3.2.2 Design of Tightening Button Behaviour.....20

3.2.3 Design of Loosening Button Behaviour.....21

3.2.4 Design of Preference Button Behaviour.....22

3.2.5 Design of Reset Button Behaviour.....23

3.3. Design of Sensor System.....23

3.4. Design of Motors System.....24

CHAPTER IV MEASUREMENT AND SYSTEM ANALYSIS

4.1. Checking The Condition of Switch System.....27

4.2. Checking Sensor System.....29

CHAPTER V CLOSING

5.1. Conclusion.....33

5.2. Suggestion.....33

References.....29

Enclosures.....30

Author’s Biography.....42

DAFTAR GAMBAR

Gambar 2.1.	Mekanisme Sensor berat.....	5
Gambar 2.2.	Sensor Berat dengan Maksimum Beban 50 Kg.....	6
Gambar 2.3.	Pengaturan Kabel Sensor.....	6
Gambar 2.4.	Jembatan Wheatstone.....	7
Gambar 2.5.	Konfigurasi full-bridge.....	8
Gambar 2.6.	Data Output, Input, Timing dan Kontrol Pemilihan Gain	9
Gambar 2.7.	Konfigurasi HX711.....	10
Gambar 2.8.	Technical Drawing JA12 N20.....	11
Gambar 2.9.	Pin pada Pro Micro.....	13
Gambar 2.10.	Modul Relay 1-channel.....	14
Gambar 3.1.	Blok diagram Sistem.....	18
Gambar 3.2.	Rangkaian Sistem Tombol.....	19
Gambar 3.3.	Desain PCB Sistem Tombol.....	19
Gambar 3.4.	Sistem Tombol pada Sepatu.....	20
Gambar 3.5.	<i>Flow Chart</i> Perilaku Tombol Pengencang.....	21
Gambar 3.6.	<i>Flow Chart</i> Perilaku Tombol Pengendur.....	22
Gambar 3.7.	Posisi Sensor pada sol.....	23
Gambar 3.8.	Motor Penggulung.....	25
Gambar 4.1.	Plot Proses Penggulungan Manual.....	27
Gambar 4.2.	Plot Proses Penggulungan Otomatis.....	28
Gambar 4.3.	Plot Proses Pengenduran manual.....	28
Gambar 4.4.	Plot Proses Pengenduran Otomatis.....	29
Gambar 4.5.	Keluaran Sensor Saat Tanpa dan dengan Beban.....	30
Gambar 4.6.	Bacaan Saat Program digabung.....	31

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penggunaan teknologi yang dapat dipakai merupakan impian orang sejak dahulu. Banyaknya film fiksi ilmiah yang memadukan teknologi dengan pakaian juga semakin populer digunakan akhir-akhir ini terutama pada film yang bertemakan *superhero*.

Pada tahun 1989 sebuah film berjudul *Back to the Future II* meramalkan bagaimana kemajuan teknologi yang terjadi pada tahun 2015. Pada film itu diramalkan pada tahun 2015 teknologi sudah maju secara pesat tapi yang paling mencolok adalah teknologi *hoverboard* dan sepatu yang dapat mengikat secara otomatis. Pada tahun 2015 perusahaan Nike meluncurkan *Nike Airmag* yang dirilis terbatas sebagai realisasi dari film tersebut. Nike Airmag memiliki tampilan yang sangat mirip dengan yang ada di film tetapi belum memiliki teknologi yang diramalkan. Pada akhir 2016 akhirnya Nike merilis *Nike Hyperadapt* yang akhirnya memiliki teknologi pengikat otomatis^[1]. Sepatu ini merupakan salah satu sepatu yang paling dicari kolektor dan merupakan sepatu dengan nilai termahal saat ini.

Saat menonton film *Back to the Future II*, penulis terinspirasi untuk membuat purwarupa dari sepatu itu. Teknologi pengikat otomatis tentu akan meningkatkan nilai dari sebuah sepatu.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan beberapa masalah, antara lain:

1. Bagaimana rancang bangun sepatu pengikat tali otomatis.
2. Bagaimana sistem pemicu yang dapat diterapkan untuk memicu pengikatan otomatis.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian pada tugas akhir ini adalah sebagai berikut:

1. Membuat rancang bangun sepatu dengan pengikat tali otomatis.
2. Membuat sistem pemicu pengikatan tali sepatu otomatis

1.4 Metodologi Penelitian

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi literatur
Pada tahap ini dilakukan pengumpulan dasar teori yang menunjang dalam penulisan tugas akhir. Dasar teori ini dapat diambil dari buku-buku, jurnal, paten, dan artikel-artikel di internet.
2. Perancangan sistem
Setelah mempelajari literatur yang ada, selanjutnya akan dilakukan perancangan sistem. Perancangan sistem terbagi sebagai berikut:
 - a. Perancangan Perangkat Keras
Cara kerja dari sistem ini menggunakan motor DC JA12-N20, sensor berat 50 KG, modul amplifier HX711, dan mikrokontroler *pro micro*.
 - b. Perancangan perangkat lunak
Perancangan perangkat lunak pada sistem ini akan dibagi menjadi tiga tahapan proses utama, yaitu:
 1. Merancang perilaku kerja motor DC berdasarkan masukan dari tombol.
 2. Membuat program awal untuk membaca masukan dari sensor berat.
 3. Menggabungkan kedua program agar keluaran dari sensor dapat memicu perilaku motor.
 - c. Melakukan *Troubleshooting*
Melakukan analisa tentang kesesuaian perilaku komponen, penyesuaian ukuran komponen terhadap ruang yang tersedia pada sepatu, dan penerapan program dengan komponen baru jika terjadi penggantian,

- d. Penarikan Kesimpulan dan Saran
Penarikan kesimpulan mengacu pada data pengujian, analisis data, dan referensi terkait. Kesimpulan menunjukkan hasil kerja secara garis besar sesuai rumusan masalah yang telah dibuat.
- e. Penulisan laporan Tugas Akhir
Tahap penulisan laporan Tugas Akhir dilakukan pada saat tahap *troubleshooting* dimulai serta setelahnya.

1.5 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari Lima Bab dengan sistematika penulisan sebagai berikut:

Bab 1: PENDAHULUAN

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.

Bab 2: DASAR TEORI

Bab ini menjelaskan tentang dasar-dasar teori yang dibutuhkan dalam pengerjaan tugas akhir ini, yang meliputi teori dasar sensor berat, penguat HX711, motor DC JA12 N20, dan mikrokontroler *pro micro*

Bab 3: PERANCANGAN SISTEM

Bab ini menjelaskan tentang perencanaan sistem perangkat keras serta perangkat lunak. Bab ini juga berisi menjelaskan tentang prosedur pengujian yang dilakukan dalam penelitian.

Bab 4: PENGUKURAN DAN ANALISIS SISTEM

Bab ini menjelaskan tentang hasil yang didapat dari pengujian tiap blok sistem secara keseluruhan.

Bab 5: PENUTUP

Bab ini menjelaskan tentang kesimpulan meliputi kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

1.6 Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Mempercepat waktu pemakaian sepatu dan membantu penyandang disabilitas.
2. Sebagai dasar penelitian lebih lanjut pada bidang *wearable-technology*, agar dapat lebih dikembangkan.

BAB II

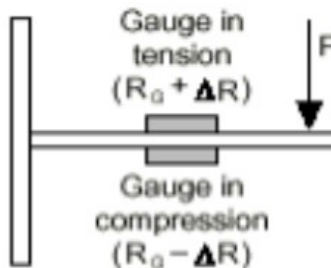
DASAR TEORI

2.1 Sensor Berat

Sensor berat atau yang biasa disebut *strain guauge* merupakan suatu sensor resistif yang berubah-ubah nilai hambatannya tergantung dari gaya yang diberikan terhadap sensor. Sensor berat memiliki berbagai macam bentuk dan ukuran yang disesuaikan dengan penerapannya.^[2]

2.1.1 Prinsip Kerja Sensor Berat

Jika gaya diaplikasikan pada sensor maka akan terjadi perubahan hambatan. Gaya yang diberikan akan membengkokkan sensor. Setiap sensor memiliki gauge factor yang umumnya bernilai sekitar 2. Gauge factor digunakan untuk mengukur berapa perubahan hambatan dari perubahan panjang dari sensor. Jika sensor mengalami perubahan sebesar $2\mu\text{ m}$ maka perubahan hambatannya adalah $2 \times 2\mu\text{ m} = 4\mu\text{ ohm}$. Perubahan hambatan dari sensor ini sangat kecil sehingga butuh penguat untuk melihat hasilnya lebih jelas.



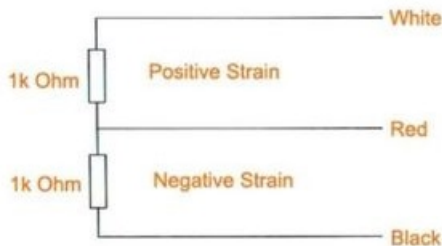
Gambar 2.1 Mekanisme Sensor berat

2.1.2 Menentukan Perilaku dari Sensor



Gambar 2.2 Sensor Berat dengan Maksimum Beban 50 Kg

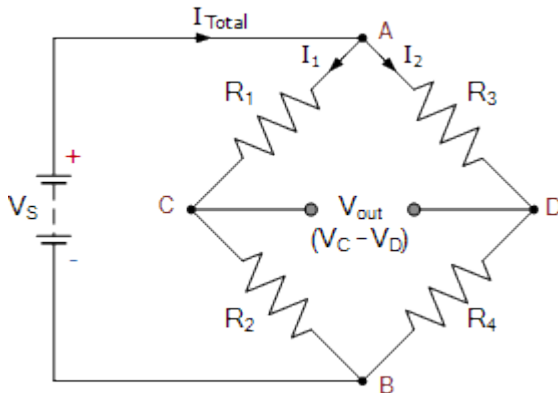
Sensor berat dengan maksimum beban 50 Kg merupakan sensor dengan tiga buah kabel yang berwarna merah, putih, dan hitam. Sensor ini memiliki positive strain dan juga negative strain sehingga pengaruh gaya dapat teraplikasi pada dua arah. Kabel hitam biasanya terhubung dengan positive strain dan kabel putih terhubung dengan negative strain. Kabel yang terhubung pada masing-masing strain bisa disebut juga sebagai pair. Kabel merah merupakan center tap. Karena tidak semua sensor memiliki konfigurasi warna yang sama maka diperlukan cara untuk menentukan mana yang merupakan pair dan mana yang center tap. Nilai hambatan pair umumnya dua kali lebih besar dari pada hambatan center tap-pair. Pada sensor 50 Kg hambatan dasarnya adalah 1K ohm jadi besar hambatan center tap dengan pair adalah 1k ohm (Merah-putih atau merah hitam) dan hambatan pair (hitam-putih) adalah 2K ohm.



Gambar 2.3 Pengaturan Kabel Sensor

2.1.3 Jembatan Wheatstone

Jembatan wheatstone merupakan salah satu cara untuk melihat perubahan signifikan dari kecilnya perubahan hambatan dari sensor. Saat hambatan seimbang maka tegangan pada tengah jembatan wheatstone akan bernilai 0 volt. Jika ada perubahan kecil pada hambatan maka terjadi ketidakseimbangan yang akan menyebabkan perubahan tegangan pada tengah jembatan. Contoh rangkaian jembatan Wheatstone ada di gambar 2.4.



Gambar 2.4 Jembatan Wheatstone

Sebagai contoh jika:

$$V_S = 5 \text{ V}$$

$$R_1 = R_3 = 1 \text{ k Ohm.}$$

$$R_2 = R_4 = 4 \text{ k Ohm.}$$

Tegangan pada titik AB sama dengan V_S yaitu 5 V. Selanjutnya mari analisa titik ACB. Pada titik ACB tegangan adalah 5 V dan total hambatan adalah 5k ohm. Maka arus I_1 adalah

$$I_1 = V_S / R_{AB}$$

$$I_1 = 5/5 \text{ k}$$

$$I_1 = 1 \text{ mA}$$

Titik ADB identik dengan ACB sehingga: $I_1 = I_2$.

Tegangan di titik A dan C adalah:

$$V_{AC} = I_1 * R_1$$

$$V_{AC} = 1\text{mA} * 1\text{k}$$

$$V_{AC} = 1 \text{ V.}$$

Karena tegangan di titik C dan D identik maka selisihnya akan sama dengan 0. Tetapi bagaimana jika posisi R2 dan R1 ditukar. Tegangan di titik C maka akan menjadi

$$V_{AC} = I_1 * R_2$$

$$V_{AC} = 4\text{k} * 1\text{mA}$$

$$V_{AC} = 4 \text{ V.}$$

Tegangan titik C berubah tetapi di titik D tetap. Maka akan ada selisih tegangan pada titik CD menjadi

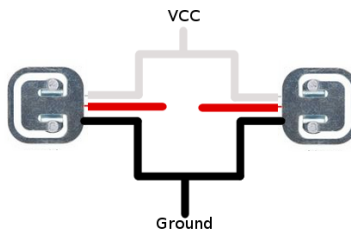
$$V_{CD} = V_C - V_D$$

$$V_{CD} = 4 - 1$$

$$V_{CD} = 3 \text{ V.}$$

Konsep inilah yang membuat jembatan wheatstone cocok untuk menganalisa sistem yang memiliki sedikit perubahan hambatan untuk dianalisa.

Pada sistem ini digunakan rangkaian full-bridge karena terdapat dua sensor yang digunakan. Konfigurasi yang dilakukan seperti pada gambar 2.5



Gambar 2.5 Konfigurasi full-bridge

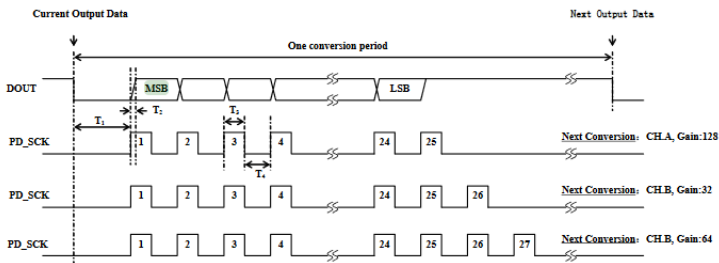
2.2 24-bit ADC HX711

HX711 merupakan ADC 24-bit yang didesain untuk aplikasi kontrol industri untuk dapat bertatap muka secara langsung dengan bridge sensor dan juga untuk timbangan badan.

Komponen ini memiliki dua channel, channel A dan B. Channel A dapat diprogram memiliki gain 128 atau 64 tergantung dari masukannya, $\pm 20\text{mV}$ atau $\pm 40\text{mV}$.

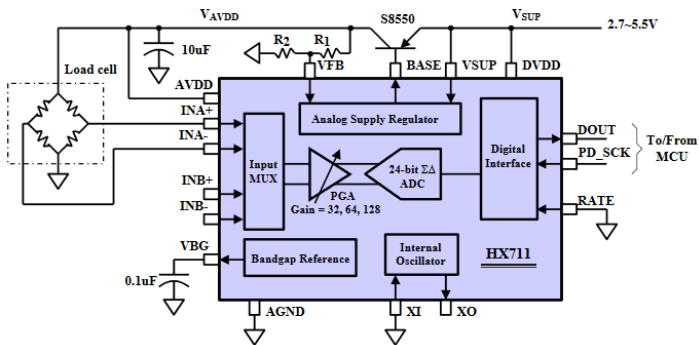
Dengan menyambungkan pin XI pada ground maka osilator internal akan aktif. Nominal output data rate saat menggunakan osilator internal adalah 10 (RATE = 0) atau 80 sampel per detik (RATE = 1). Jika keluaran data yang akurat dibutuhkan maka clock eksternal atau crystal dapat digunakan. Sebuah crystal dapat dihubungkan pada pin X1 dan X0 sedangkan clock eksternal dapat terhubung ke pin X1 melalui kapasitor 20pF yang berpasangan dengan AC. Saat crystal eksternal digunakan maka osilator internal secara otomatis akan tidak aktif.

Pin PD_SCK dan DOUT berfungsi sebagai penerima data, pemilihan input, pemilihan gain, dan kontrol power down. Saat data keluaran belum siap untuk diambil, pin DOUT akan high dan input PD_SCK low. Saat DOUT berubah menjadi low maka itu menandakan data sudah siap diambil. Dengan memberikan 25 hingga 27 pulsa clock positif pada pin PD_SCK, data akan bergeser dari pin output DOUT. Setiap pulsa PD_SCK akan menggeser satu bit dimulai dari MSB (Most Significant Bit) hingga seluruh 24 bit bergeser semua. Bit ke 25 pada input PD_SCK akan membuat pin DOUT kembali high.



Gambar 2.6 Data Output, Input, Timing dan Kontrol Pemilihan Gain

Pada datasheet diberikan petunjuk pemrograman driver. Terdapat berbagai modifikasi program driver ini untuk memudahkan penggunaan dan memaksimalkan fungsi dari komponen ini. Pada arduino sudah terdapat banyak *library* yang sangat membantu dalam pemrograman komponen ini.



Gambar 2.7 Konfigurasi HX711^[3]

2.3 Motor DC JA12 N20

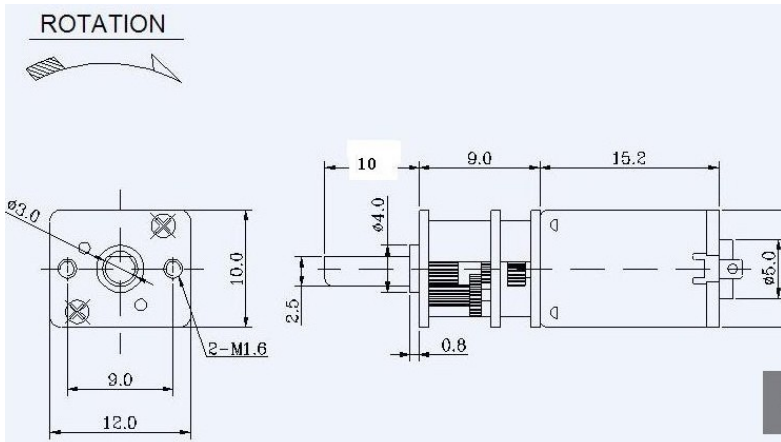
Motor DC memiliki dua bagian dasar yaitu stator dan rotor. Stator atau bagian stationer adalah bagian yang bersifat diam pada badan motor. Stator berfungsi menghasilkan medan magnet yang dibangkitkan dari sebuah koil (elektromagnetik) atau magnet permanen. Bagian stator terdiri dari bodi motor yang memiliki magnet pada keseluruhan permukaan bodi. Untuk motor berukuran kecil, magnet tersebut adalah magnet permanen sedangkan untuk motor berukuran besar maka akan menggunakan koil atau kumparan yang dililitkan pada lempeng-lempeng besi. Rotor adalah sebuah bagian berputar pada motor DC. Sedangkan rotor berupa sebuah koil yang dialiri oleh arus listrik. Suatu kumparan motor akan berfungsi apabila mempunyai:

1. Kumparan medan, berfungsi sebagai penghasil medan magnet
2. Kumparan jangkar, berfungsi sebagai pengimbas GGL pada konduktor yang terletak pada alur-alur jangkar.
3. Celah udara yang memungkinkan berputarnya jangkar dalam medan magnet.

Motor DC JA12 N20 merupakan salah satu motor DC yang berukuran kecil tetapi memiliki torsi yang cukup tinggi. Motor ini sudah memiliki sistem penguncian sehingga saat tidak di-supply energi maka posisinya akan tetap walau kita berusaha mengubahnya dengan memutar porosnya. Motor ini dapat diberikan tegangan hingga 12 V.

Motor ini memiliki berbagai macam pilihan rasio gir. Rasio ini menentukan kecepatan putaran dari motor. Semakin kecil rasionya semakin kecil pula. Rasio gir yang digunakan pada sistem ini adalah

1:298. Pada tegangan 12 V akan menghasilkan 100 RPM dan torsi 16 Kg cm. Pada tegangan 6 V akan menghasilkan 50 RPM dan torsi 10 Kg cm dan pada tegangan 3 V akan menghasilkan 25 RPM dan torsi 4 Kg cm. Motor ini memiliki torsi yang cukup besar walaupun ukurannya sangat kecil. Bahannya yang sepenuhnya terbuat dari metal membuatnya juga semakin kuat.



Gambar 2.8 Technical Drawing JA12 N20

2.3.1 Analisa Motor DC

Sebuah rangkaian ekivalensi dari motor DC secara single link dapat didefinisikan dalam sebuah persamaan, yaitu:

$$\frac{L_a di_a}{dt} + R_a i_a = V - V_b \quad (2.1)$$

Persamaan gerak untuk motor didefinisikan sebagai:

$$\frac{J_m d^2 \theta_m}{dt^2} + \frac{B_m d\theta_m}{dt} = \tau_m - \frac{1}{n} \tau_1 \quad (2.2)$$

Hubungan antara torsi motor (τ_m) dengan arus jangkar I_a dinyatakan dengan persamaan :

$$\tau_M = Kt * I_a \quad (2.3)$$

K_t : konstanta torsi (N.m/Amp)

Hubungan antara tegangan terminal (V_t) dan kecepatan sudut motor (ω_m) dinyatakan dalam dengan persamaan:

$$V_t = K_e \cdot \omega_m \quad (2.4)$$

K_e : konstanta GGL balik (V.sec/rad)

Dengan mensubstitusikan persamaan 2.4 ke dalam persamaan 2.1 maka akan dihasilkan persamaan dalam domain s sebagai berikut:

$$(L_a s + R_a) I_a(s) = V(s) - K_e \omega_m(s) \quad (2.5)$$

Dengan mensubstitusikan persamaan 2.3 dengan persamaan 2.2 maka akan dihasilkan persamaan dalam domain s sebagai berikut :

$$(J_m s + B_m) \omega_m(s) = K_T I_a(s) - \frac{1}{n} \tau_1(s) \quad (2.6)$$

Dengan menganggap bahwa torsi beban bernilai nol, maka dengan menggabungkan 2.5 dan 2.6 akan didapatkan *transfer function* tegangan terminal motor (V_t) terhadap kecepatan sudut motor yang dirumuskan dalam persamaan berikut:

$$\frac{\omega_m(s)}{V_t} = \frac{K_t}{(L_a s + R_a)(J_m s + B_m) K_e K_T} \quad (2.7)$$

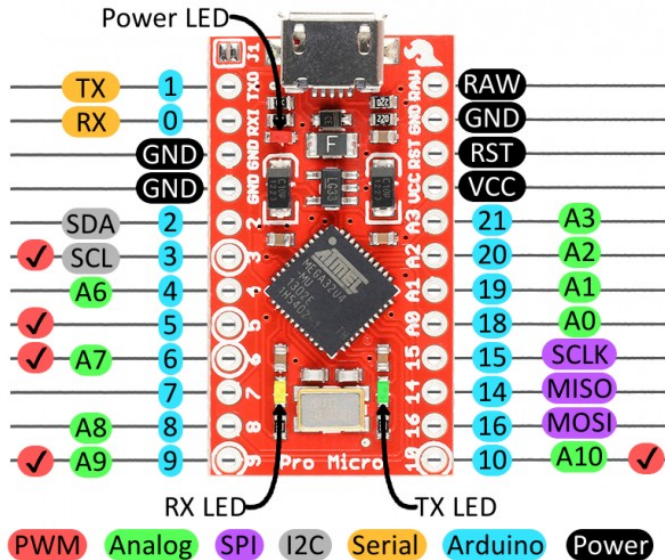
Dengan melihat persamaan 2.7 maka dapat diperoleh diagram blok dari motor DC. Transfer function motor DC berfungsi untuk menyatakan model matematika motor secara menyeluruh

2.4 Mikrokontroller Pro Micro

Pro Micro merupakan salah satu mikrokontroler berukuran kecil yang compatible dengan arduino. Kontroler ini menggunakan prosesor ATmega32U4.

Semua pin I/O dan daya pada pro micro diposisikan secara paralel dan saling berhadapan. Beberapa pin berfungsi sebagai input atau output daya, yang lainnya berfungsi sebagai pin I/O. Pin I/O memiliki

kemampuan spesial misalnya sebagai input analog. Gambar 2.6 menunjukkan fungsi dari setiap pin



Gambar 2.9 Pin pada Pro Micro

Berikut penjelasan beberapa pin daya

- RAW adalah input tegangan *unregulated* untuk pro micro. Jika diberi daya melalui USB maka tegangan pada pin ini sekitar 4,8 V (5V dari USB minus tegangan drop dari dioda schottkey). Jika diberi daya externa; melalui pin ini tegangan yang dapat diterapkan mencapai 12V
- VCC adalah tegangan yang disuplai ke Atmega32U4. Tegangan ini tergantung dari versi mana yang digunakan, bernilai 3,3V jika menggunakan 3.3V/8MHz pro micro dan 5V jika menggunakan 5V,16MHz pro micro. Tegangan ini teregulasi oleh tegangan yang diterapkan pada pin RAW. Jika kontroler diberi daya melalui pin RAW atau USB, maka pin ini dapat berfungsi sebagai output untuk menyuplai perangkat lain.

- RST dapat digunakan untuk mereset kontroler. Pin ini “pulled high” dengan resistor 10K Ohm yang terdapat pada board. Pin ini bersifat active-low sehingga untuk mengaktifkannya harus dihubungkan menuju ground. Pro micro akan tetap dalam kondisi mati hingga pin reset kembali bernilai high
- GND merupakan ground dari sistem.

Pro micro memiliki 18 pin I/O. Semuanya dapat berfungsi sebagai input atau output digital. Pin ini penomorannya berdasarkan Arduino IDE via nilai integer antara 0-21.

Sembilan pin dapat berfungsi sebagai ADC dan dapat digunakan sebagai input analog. Pin ini berguna untuk membaca potensiometer atau perangkat analog lain menggunakan fungsi “analogRead([pin]).

Terdapat lima pin yang dapat berfungsi sebagai PWM yang dapat digunakan sebagai output analog dengan menggunakan fungsi “analogWrite([pin], [nilai])”. Pin serial, I2C, dan SPI juga tersedia pada kontroler ini.

2.5 Modul Relay

Digunakan 2 1-channel relay pada proyek ini, dengan tegangan 3.3V/5V DC untuk melakukan kontrol terhadap relay. Sangat baik untuk melakukan switch pada perangkat AC maupun DC yang membutuhkan arus dan tegangan yang besar.

Relay yang digunakan ialah SPDT atau single pole double throw dengan arus dan tegangan maksimal 10A/250V AC. Ground pada koil relay terpisah dengan ground pada sinyal input. Namun keduanya dapat disatukan dengan memberikan jumper pada header.



Gambar 2.10 Modul Relay 1-channel

Pada relay terdapat 3 port output. Common, normally closed, dan normally open. Saat relay dalam keadaan tidak aktif common akan terhubung dengan port normally closed tapi terputus dengan normally open dan saat keadaan aktif common akan terhubung dengan normally open tetapi terputus dengan normally closed. Keadaan ini memungkinkan relay digunakan sebagai kontroler H-bridge.

BAB III

PERANCANGAN SISTEM

Perancangan sistem meliputi perangkat keras dan perangkat lunak. Perangkat keras yang digunakan yaitu Mikrokontroler Pro Micro, Sensor Berat dengan beban maksimum 50 KG, 24-bit ADC HX711, Motor DC JA12 N20, dan modul relay.

Perangkat lunak meliputi program mikrokontroler dengan menggunakan perangkat lunak Arduino. Program ini akan mengontrol perilaku dari motor berdasarkan masukan yang diberikan.

Terdapat tiga tombol yang berfungsi untuk mengontrol perilaku motor DC. Terdapat tiga buah tombol berfungsi sebagai pengatur kekencangan, tombol pengencang yang berfungsi mengencangkan tali, tombol pengendur yang berfungsi untuk mengendurkan dan apabila ditekan dua kali secara cepat akan mengendurkan tali ke pengenduran maksimal, dan tombol otomatis yang akan mengencangkan atau mengendurkan tali menuju posisi kekencangan optimal. Terdapat juga satu cara untuk memicu pengencangan yaitu menggunakan hentakan. Terdapat dua sensor berat yang dipasang pada sol sepatu. Sensor tersebut akan memberi masukan nilai ke prosesor mikrokontroler yang apabila mencapai nilai tertentu akan memicu pengikatan otomatis menuju kekencangan optimal.

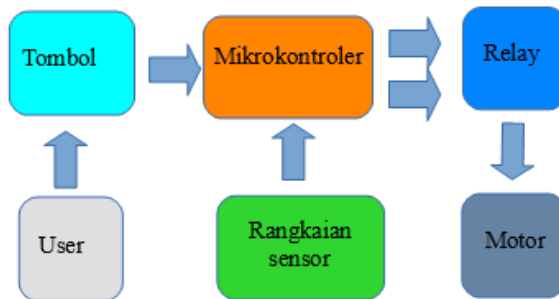
3.1 Diagram Blok Sistem

Pada sistem rancangan ini, terdapat dua sumber pemicu dari perilaku motor. Yang pertama adalah menggunakan tombol yang dapat dikontrol oleh user. Yang kedua adalah dengan menggunakan masukan dari sensor berat yang sudah terpasang pada bagian sol. Terdapat tiga tombol yang berfungsi untuk mengontrol perilaku motor. Tombol pengencang, pengendur, dan otomatis. Jika tombol pengendur ditekan dua kali secara cepat maka akan dilakukan pengenduran maksimal. Tombol otomatis akan mengarahkan motor menuju kekencangan optimal. Jika posisi tali lebih kendur dari pengencangan maka akan dikencangkan dan apabila lebih kencang maka akan dikendurkan.

Pada rangkaian sensor, keluaran hanya diberikan jika dilakukan hentakan posisi tali ada di kendur maksimum. Jika hentakan dilakukan pada saat tali tidak berada pada posisi kendur maksimum maka tidak akan terjadi pengencangan. Sistem ini baik digunakan pada saat awal

pemakaian karena posisi tali biasanya sudah dalam posisi pengenduran maksimum.

Rangkaian relay dengan motor dipasang dengan konfigurasi H-bridge sehingga dapat bergerak searah dan berlawanan jarum jam tergantung dari masukannya. Parameter pergerakan motor menggunakan basis waktu (dalam milisecond) sehingga parameter kekencangan pada sistem ini akan menggunakan waktu sebagai acuannya.



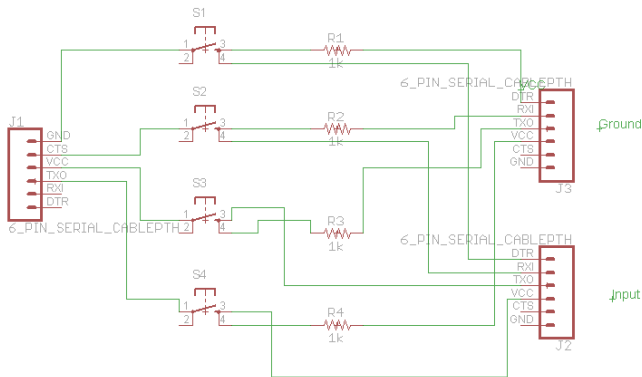
Gambar 3.1 Blok diagram Sistem

3.2 Perancangan Sistem Tombol

Sistem tombol merupakan sistem yang berfungsi sebagai masukan yang dikontrol oleh pengguna. Terdapat empat tombol yang akan digunakan. Tombol pengencang dan pengendur bekerja berdasarkan dengan keadaannya sekarang dan juga keadaan sebelumnya. Tombol preferensi untuk menyimpan tingkat kekencangan yang nyaman oleh pengguna. Dan tombol reset untuk melakukan reset agar masukan dari sensor dapat digunakan sebagai pemicu.

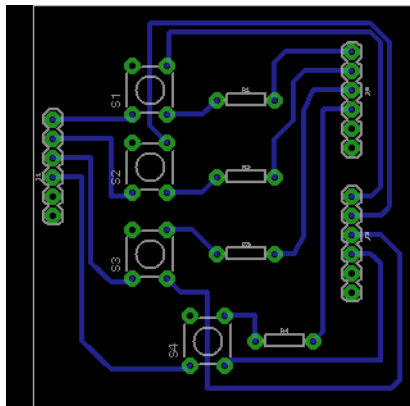
3.2.1 Perancangan Perangkat Keras

Perangkat keras terdiri atas mikrokontroler pro micro, dan empat buah push button. Berikut adalah rangkaian dari sistem tombol :



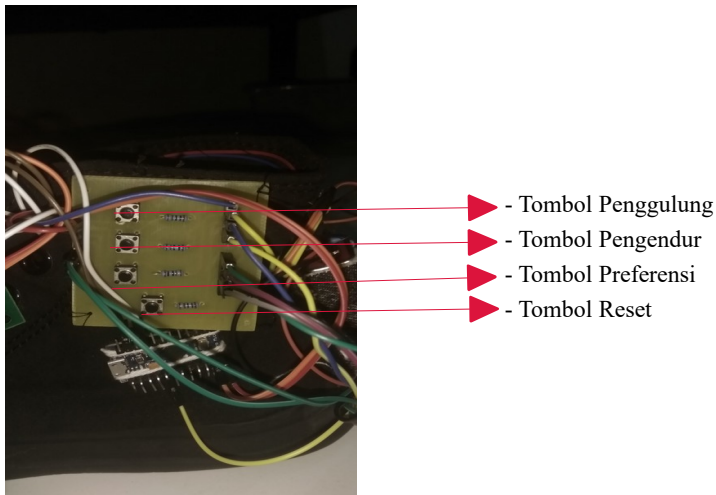
Gambar 3.2 Rangkaian Sistem Tombol

Seluruh tombol lalu ditempatkan pada PCB. Gambar 3.3 menunjukkan hasil desain PCB :



Gambar 3.3 Desain PCB Sistem Tombol

Switch 1 berfungsi sebagai tombol pengencang. Switch 2 berfungsi sebagai tombol pengendur. Switch 3 berfungsi sebagai penyimpan kekencangan referensi, switch 4 berfungsi sebagai reset pembacaan dari sistem sensor. Board ini diletakkan pada bagian luar dari sepatu.



Gambar 3.4 Sistem Tombol pada Sepatu

3.2.2 Perancangan Perilaku Tombol Pengencang

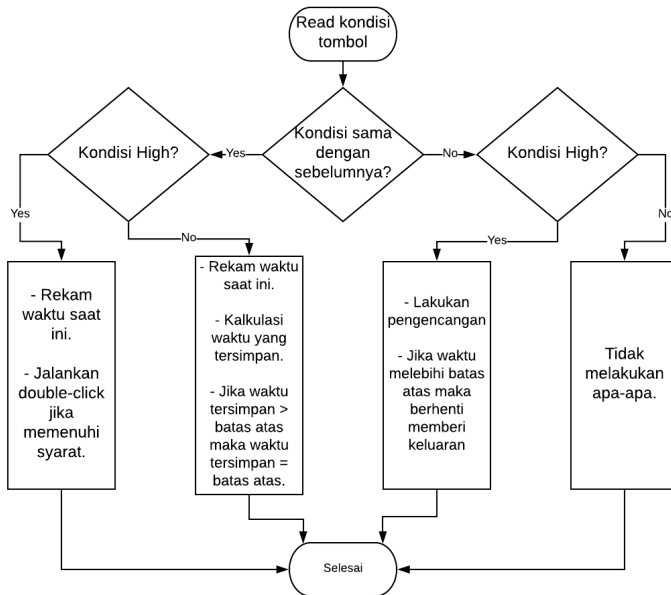
Tombol ini bertanggung jawab sebagai pengencang dari motor. Berikut rincian perilaku yang dilakukan tombol ini.

Jika kondisi saat ini sama dengan kondisi sebelumnya, maka akan dilihat lagi bagaimana kondisi tombol. Jika posisi high maka akan direkam waktunya saat tombol dipencet. Dan jika tidak (kondisi saat melepas tombol setelah sebelumnya tombol ditekan) maka akan dilakukan kalkulasi dari waktu akhir. Jika waktu tersimpan lebih besar dari batas atas maka waktu tersimpan akan diubah menjadi batas atas.

Jika kondisi saat ini sama dengan kondisi sebelumnya, maka akan dilihat lagi bagaimana kondisi tombol. Jika kondisi high (menahan tekanan tombol) maka terjadi proses penggulungan. Proses penggulungan akan berhenti jika waktu penggulungan lebih besar dari

batas atas. Jika kondisi low (Tidak ditekan sama sekali) maka tidak dilakukan penggulungan.

Double-click akan memicu terjadinya pengencangan maksimum. Syarat *double-click* adalah jika waktu jarak tekan antara penekanan pertama dan kedua kurang dari 700 ms.



Gambar 3.5 *Flow Chart* Perilaku Tombol Pengencang

3.2.3 Perancangan Perilaku Tombol Pengendur

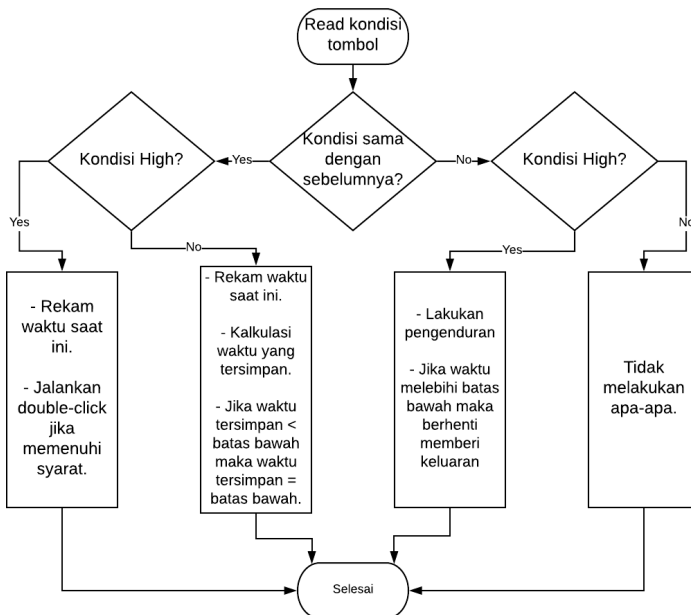
Tombol ini bertanggung jawab sebagai pengencang dari motor. Berikut rincian perilaku yang dilakukan tombol ini.

Jika kondisi saat ini sama dengan kondisi sebelumnya, maka akan dilihat lagi bagaimana kondisi tombol. Jika posisi high maka akan direkam waktunya saat tombol dipencet. Perintah dengan pemicu *double-click* juga dilakukan jika kondisinya terpenuhi. Dan jika tidak (kondisi saat melepas tombol setelah sebelumnya tombol ditekan) maka akan dilakukan kalkulasi dari waktu akhir. Jika waktu tersimpan lebih

besar dari batas bawah maka waktu tersimpan akan diubah menjadi batas bawah.

Jika kondisi saat ini sama dengan kondisi sebelumnya, maka akan dilihat lagi bagaimana kondisi tombol. Jika kondisi high (menahan tekanan tombol) maka terjadi proses pengenduran. Proses pengenduran akan berhenti jika waktu pengenduran lebih besar dari batas bawah. Jika kondisi low (Tidak ditekan sama sekali) maka tidak dilakukan penggulungan.

Double-click akan memicu terjadinya pengenduran maksimum. Syarat *double-click* adalah jika waktu jarak tekan antara penekanan pertama dan kedua kurang dari 700 ms.



Gambar 3.6 *Flow Chart* Perilaku Tombol Pengendur

3.2.4 Perancangan Perilaku Tombol Preferensi

Tombol preferensi akan menyimpan waktu yang saat ini tersimpan menjadi waktu preferensi. Penggulungan otomatis akan menggerakkan

motor menuju waktu preferensi. Nilai preferensi awal adalah 200 ms dan dapat diubah sesuai kenyamanan pemakai. Nilai preferensi tidak dapat lebih besar dari batas atas. Saat tombol ditekan maka nilai waktu sekarang akan disimpan menjadi waktu preferensi.

3.2.5 Perancangan Perilaku Tombol Reset

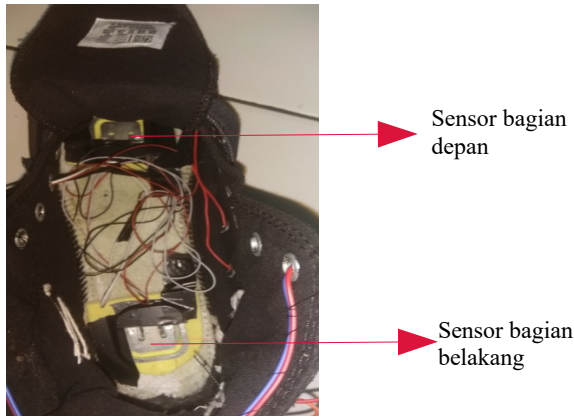
Tombol reset akan melakukan reset agar masukan dari sensor dapat digunakan sebagai pemicu pengencangan. Pada sistem sensor, jika tekanan diberikan maka akan diberikan perintah untuk melakukan pengencangan ke preferensi. Saat sepatu digunakan sensor akan selalu membaca adanya tekanan sehingga pengencangan akan selalu dilakukan padahal fungsi ini umumnya hanya digunakan sekali saat awal pemakaian sepatu. Maka dibuatlah parameter tambahan agar masukan dari sensor hanya dapat digunakan sekali saat pemakaian awal.

Sebuah koefisien dengan nilai 0 ditambahkan pada parameter. Pengencangan dari sensor hanya akan dilakukan jika nilai koefisien adalah 0 dan terjadi tekanan pada sensor. Saat terjadi pengencangan nilai koefisien akan ditambah 1 sehingga jika sensor tetap membaca adanya tekanan maka pengencangan tidak akan dilakukan karena nilai koefisien sudah tidak sama dengan 0.

Tombol reset ini akan mengubah nilai koefisien kembali menjadi 0. Sebuah LED digunakan sebagai indikator nilai koefisien. Jika koefisien = 0 maka LED akan menyala dan jika nilai koefisien tidak sama dengan 0 maka LED akan mati.

3.3 Perancangan Sistem Sensor

Pada sistem ini digunakan dua buah sensor berat. Sensor ini akan ditanam pada bagian sol di bagian depan dan belakang dan akan menjadi pemicu dari aktuator.^[4]



Gambar 3.7 Posisi Sensor pada sol

Sensor akan dikonfigurasi menggunakan konsep jembatan wheatstone. Kabel hitam kedua sensor akan saling terhubung dan kabel putih kedua sensor juga akan saling terhubung. Kabel merah akan menjadi masukan yang nanti akan dikuatkan oleh penguat. Kabel hitam dan putih akan terhubung ke port eksitasi dan kedua kabel merah terhubung ke A- dan A+. Setelah melalui proses penguatan maka nilai analog akan dikirim menuju prosesor untuk diproses.

Penggunaan penguat sangat diperlukan karena perubahan tahanan dari sensor sangat kecil. Berdasarkan eksperimen yang dilakukan tahanan hanya berubah 1 ohm jika diberi beban sekitar 10 Kg.

Setelah sensor terhubung dengan penguat maka keluaran dari penguat dapat dipanen pada mikrokontroler. Nilai mentah dari penguat tidak terlalu konsisten karena noise dari sensor juga ikut dikuatkan misalnya dari gangguan elektromagnetik dan juga suhu.

3.4 Perancangan Sistem Motor

Motor DC merupakan penggulung tali dan juga aktuator utama. Dua buah relay akan mengatur arah putaran dari motor. Motor dan relay akan menggunakan konfigurasi H-Bridge^[5]. Sumber positif dari baterai 9 v akan terpasang pada normally closed dari modul relay, sumber negatif akan terpasang pada normally opened dan kabel menuju motor akan terhubung ke common. Saat terjadi pengencangan relay B akan aktif sehingga motor akan melakukan penggulungan karena motor mendapatkan arus dari sumber positif menuju negatif. Saat proses

pengenduran dilakukan, relay A akan aktif sehingga motor akan melakukan pengenduran karena motor mendapatkan arus dari sumber negatif menuju negatif. Tombol pengencang akan mengaktifkan relay B dan tombol pengendur akan mengaktifkan relay A.



Gambar 3.8 Motor Penggulung

BAB IV

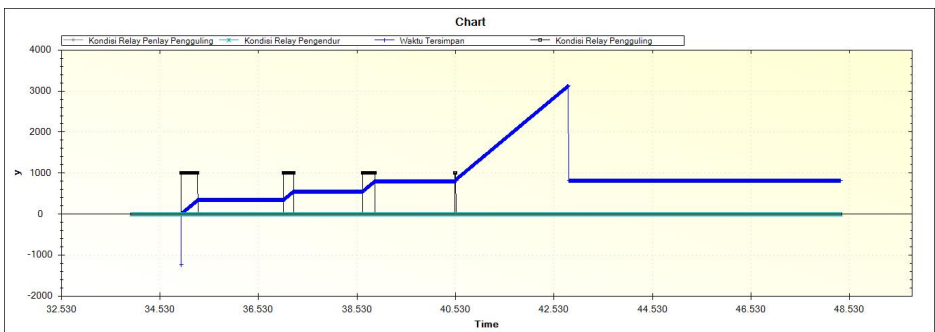
PENGUKURAN DAN ANALISIS SISTEM

4.1 Pemeriksaan Kondisi Sistem Tombol

Pemeriksaan sistem tombol dilakukan agar seluruh masukan dari pengguna dapat diterima dengan baik oleh mikro kontroler. Proses pemeriksaan dilakukan dengan melakukan berbagai masukan dan kondisi yang dapat dilakukan oleh tombol. Waktu pengencangan dapat dilihat melalui serial monitor dari IDE arduino. Waktu tersimpan adalah plot biru. Output penggulangan adalah plot hitam dan output pengenduran adalah plot cyan.

Pada sistem tombol pengencang semua berjalan dengan baik. Proses penggulangan juga sudah berjalan dengan lancar (Gambar 4.1). Proses pengubahan waktu jika waktu gulung lebih dari yang seharusnya juga sudah berjalan dengan baik seperti yang terlihat pada detik 41. Waktu tersimpan merangkak naik terus tetapi output sudah tidak aktif lagi. Saat tombol dilepas maka waktu tersimpan akan kembali ke batas maksimal (820 ms) seperti yang terlihat pada grafik.

Sebuah masalah ditemukan pada proses penggulangan yaitu kurangnya kemampuan motor untuk menggulung tali sepenuhnya. Motor mengalami stall saat mencapai kekencangan tertentu. Hal ini terjadi karena torsi motor yang kurang besar.



Gambar 4.1 Plot Proses Penggulangan Manual

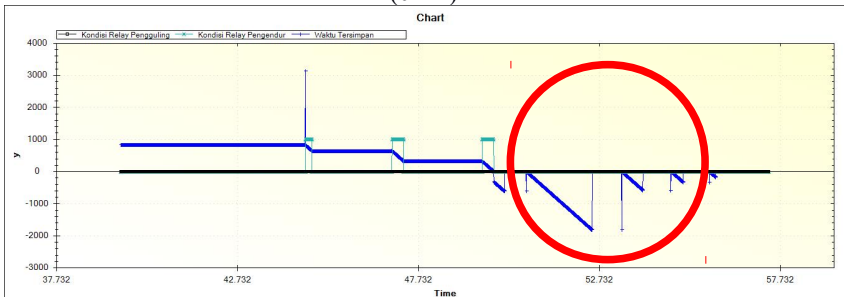
Pengencangan otomatis juga sudah berjalan dengan baik seperti yang terlihat pada gambar 4.2. Saat tombol dua kali ditekan maka

penggulungan akan dilakukan menuju kekencangan preferensi (kekencangan preferensi yang digunakan adalah 624ms).



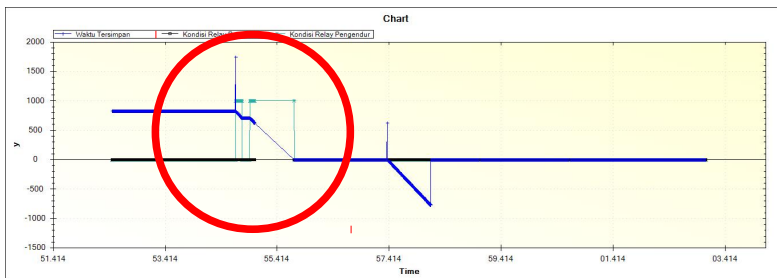
Gambar 4.2 Plot Proses Penggulungan Otomatis

Pada sistem tombol pengendur semua berjalan dengan baik. Proses pengenduran juga sudah berjalan dengan lancar (Gambar4.3). Saat tombol pengendur ditekan maka waktu tersimpan akan berkurang. Proses perubahan waktu jika waktu kendur lebih dari yang seharusnya juga sudah berjalan dengan baik seperti yang terlihat pada bagian yang dilingkari. Pengurangan waktu terus terjadi tetapi output pengenduran tidak aktif dan saat tombol dilepas maka waktu tersimpan akan kembali ke batas kekenduran maksimum (0 ms).



Gambar 4.3 Waktu Proses Pengenduran Manual

Proses pengenduran otomatis juga sudah berjalan dengan baik. Saat tombol pengendur ditekan dua kali maka dilakukan pengenduran menuju 0ms seperti yang terlihat pada gambar 4.4



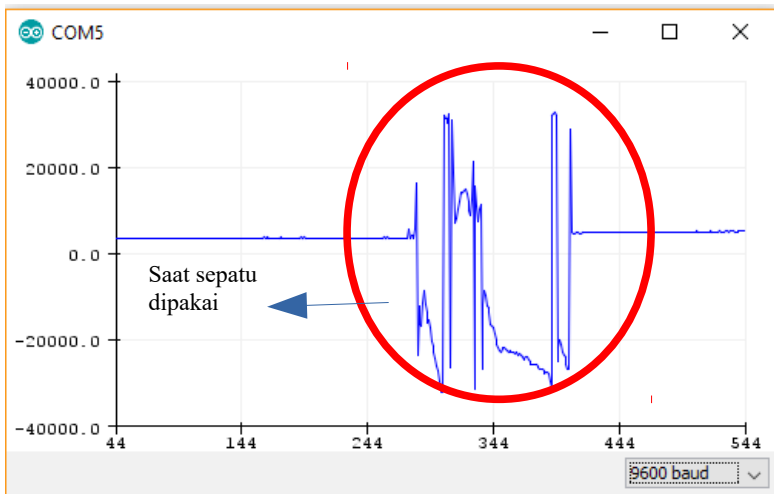
Gambar 4.4 Plot Proses Pengenduran Otomatis

Pada sistem preferensi juga sudah berjalan dengan baik. Saat tombol ditekan maka waktu saat itu yang tersimpan akan diubah menjadi waktu preferensi.

Tombol reset juga sudah berjalan seperti seharusnya. Saat tombol ditekan maka nilai koefisien akan berubah menjadi 0 dan LED menyala. Pada fase ini sistem pengencangan otomatis dengan sensor sebagai pemicu sudah siap lagi digunakan.

4.2 Pemeriksaan Sistem Sensor

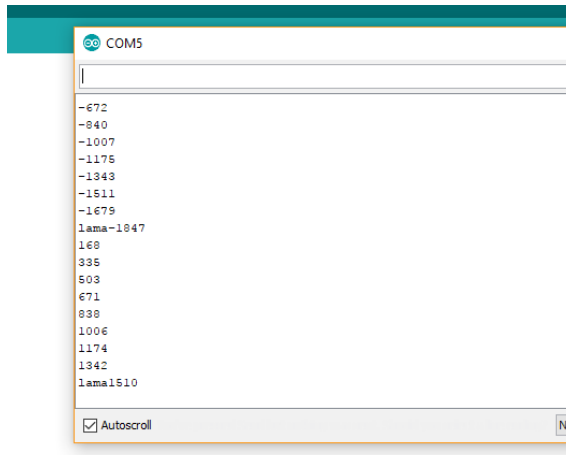
Hasil keluaran dari HX711 merupakan nilai mentah dari sensor. Nilai sensor jika tidak ada beban walau berubah-ubah setiap saat tetapi masih memiliki nilai yang cukup konsisten. Nilai tanpa beban kadang dapat berbeda jauh dari biasanya setelah sensor diberi beban. Umumnya nilai tanpa beban pada sensor berada pada 8000-35000. Saat sepatu dipakai nilai dari sensor akan negatif. Saat beberapa lama sepatu digunakan kadang nilai sensor akan berubah menjadi positif. Hal ini terjadi karena posisi telapak kaki tidak selalu menempel pada sensor sehingga sensor akan membacanya sebagai tidak adanya beban.



Gambar 4.5 Keluaran Sensor Saat Tanpa dan dengan Beban

Sejak awal hingga perancangan semuanya menggunakan program yang terpisah dengan menggunakan dua program berbeda, program pengulungan dan program pembacaan sensor. Tetapi setelah kedua program digabungkan terjadi masalah. Sistem sensor masih berjalan dengan baik tetapi sistem tombol mengalami masalah.

Respon keluaran terhadap tombol tidak langsung terjadi. Terjadi delay yang cukup lama dari saat tombol ditekan hingga output merespon. Saat tombol ditekan sebentar pembacaan sudah mulai berjalan tetapi keluaran tidak melakukan keluaran. Hal ini membuat sistem tidak berjalan dengan baik.



Gambar 4.6 Bacaan Saat Program digabung

Hal ini terjadi karena sistem sensor meminta supply serial clock dari kontroler. Berdasarkan datasheet, untuk menghasilkan satu kali bacaan dibutuhkan 24 periode pulsa dari serial clock sehingga seluruh masukan sistem dari mengalami delay. Pada akhirnya digunakan dua kontroler pada seluruh sistem. Yang pertama untuk sistem tombol dan kedua untuk sistem sensor.

BAB V PENUTUP

5.1 Kesimpulan

Pada tugas akhir ini telah dirancang sebuah purwarupa pengikat tali sepatu otomatis. Alat ini menggunakan tombol dan sensor berat sebagai pemicu dari pengikatan. Sensor berat digunakan sebagai pemicu pengikatan otomatis dan tombol digunakan sebagai pemicu pengikatan manual dan juga otomatis. Sebuah tombol digunakan sebagai pemicu pengencangan dan tombol lain sebagai pemicu pengenduran. Sebuah tombol preferensi juga diterapkan untuk menentukan kekencangan preferensi dari pengguna. Mikrokontroler pro micro digunakan sebagai pengatur seluruh sistem. HX711 berfungsi sebagai penghubung antara sensor berat dengan mikrokontroler dan juga berfungsi untuk menguatkan nilai dari sensor.

Berdasarkan pengujian dan analisis data, maka dapat ditarik kesimpulan bahwa penggunaan library HX711 juga tidak dapat dilakukan bersamaan dengan sistem yang membutuhkan respon keluaran dengan real-time karena akan membuat respon mengalami gangguan. Penggunaan motor JA12 N20 juga kurang cocok karena torsi yang diberikan terlalu kecil. Untuk hasil yang maksimal motor dengan torsi lebih dari 25 KG cm sebaiknya digunakan.

5.2 Saran

1. Penggunaan sensor yang berbentuk lembaran piezoelectric karena sensor dengan bahan metal seperti yang digunakan lebih cocok digunakan pada permukaan yang keras dan padat.^[6]
2. Penggunaan motor dengan torsi yang lebih besar
3. Pemasangan sistem penguncian jika motor yang digunakan memiliki torsi yang kecil

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Beers, TA, Friton, MR & Hatfield, TL 2014, Automatic Lacing System, US8769844 B2
- [2] Choi I, Ricci C. (1997). Foot-mounted gesture detection and its application in virtual environments. 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, Oct. 1997
- [3] 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales, https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf. 20 November 2017 2017.
- [4] Alt, J, Kimball, J. 2017, Device for automatically tightening and loosening laces, US9578926 B2
- [5] <URL : <https://www.electro-tech-online.com/threads/how-to-turn-a-5v-dc-motor-both-clockwise-and-anticlockwise.143686/>> 3 Desember 2017
- [6] Per Hellstrom, Mia Folke, Martin, Ekström. (2015) Wearable Weight Estimation System. Procedia Computer Science Vol. 64, September 2015, pp. 146-152.

Lampiran

Kode Sistem Tombol Saja

```
int baca = 0;
int bacax = 0;
int tombol = 5;
int tombol1 = 7;
int LamaPencet = 0;
int LamaPencet1 = 0;
int kondisiawal = 0;
int kondisiawal1 = 0;
int kondisi= 0;
int kondisi1= 0;
int mulai = 0;
int mulai1 = 0;
int putus = 0;
int putus1 = 0;
int tahan = 0;
int tahan1 = 0;
int lamaSebelum = 0;
int lamaSebelum1 = 0;
int sensor = 0;
int kenceng = 0;
int kendor = 0;
int swic = 8;
int lampu = 18;
int lampu1 = 19;
int tres = 200;
int condi = 0;
int condiawal = 0;
int waktucet = 0;
int kadal = 0;
int kadal1 = 0;
int atas = 820;
int bawah = 0;
```



```

int waktuon = 0;
int waktuona = 0;
int darisen = 14;
int koe = 0;
int masuk = 4;
int indikator = 9;
int koex = 0;

```

```

void setup() {
    // put your setup code here, to run once:
    pinMode(tombol,INPUT);
    pinMode(tombol1,INPUT);
    pinMode (swic, INPUT);
    pinMode (darisen, INPUT);
    pinMode (lampu, OUTPUT);
    pinMode (lampu1, OUTPUT);
    pinMode (masuk, INPUT);
    pinMode (indikator, OUTPUT);
    Serial.begin(9600);
}
void loop() {
    // put your main code here, to run repeatedly:
    // Inisiasi kondisi input
    kondisi = digitalRead(tombol);
    kondisi1 = digitalRead(tombol1);
    condi = digitalRead(swic);
    sensor = digitalRead (darisen);

```

```

    if (kondisi != kondisiawal) {
        // Jika baru dipencet
        if (kondisi == HIGH) {
            mulai = millis();
            if (mulai - waktucet < 700 && waktucet != 0 && LamaPencet <
tres) {
                kenceng = kenceng +1;
            }
        }
    }
}

```

```

// Jika baru dilepas
else {
    if (kenceng > 0){
        digitalWrite (lampu,1);
        delay (tres-LamaPencet);
        LamaPencet = tres;
        kenceng = 0;
        Serial.print ("lama ");
        Serial.println(LamaPencet);
    }
    else {

        putus = millis();
        LamaPencet = lamaSebelum + putus - mulai;
        Serial.print ("lama");
        Serial.println (LamaPencet);
        if (LamaPencet > atas) {
            LamaPencet = atas;
        }
    }
}
}
}
// Jika tombol ditahan
else {
    if (kondisi == HIGH){
        tahan = millis();
        kadal = lamaSebelum + tahan - mulai;
        Serial.println (kadal);
        digitalWrite (lampu,1);
        if (kadal > atas){
            digitalWrite (lampu,0);
            LamaPencet = atas;
        }
    }
}
// Jila tidak dipencet
else {
    digitalWrite (lampu,0);

```

```

    }
    }
    // TOMBOL KEDUA
    if (kondisi1 != kondisiawal1) {
// Jika baru dipencet
    if (kondisi1 == HIGH) {
        mulai1 = millis();
        if (mulai1 - waktuon < 700 && waktuon != 0 && LamaPencet >
bawah) {
            kendor = kendor +1;
        }
        if (kondisi1 == LOW && kondisiawal1 == HIGH) {
            digitalWrite (lampu1,0);
        }
    }
// Jika baru dilepas
    else {
        if (kendor >0){
            digitalWrite (lampu1,1);
            delay (LamaPencet);
            LamaPencet = 0;
            kendor = 0;
            Serial.print ("lama ");
            Serial.println(LamaPencet);
        }
        else {

            putus1 = millis();
            LamaPencet = lamaSebelum + mulai1 - putus1;
            Serial.print ("lama");
            Serial.println (LamaPencet);
            if (LamaPencet < 0) {
                LamaPencet = 0;
            }
        }
    }
}
}
}
// Jika tombol ditahan

```

```

else {
  if (kondisi1 == HIGH){
    tahan1 = millis();
    kadall = lamaSebelum + mulai1 - tahan1;
    Serial.println (kadall);
    if (kadall < bawah){
      digitalWrite (lampu1,0);
      LamaPencet = 0;
    }
    else {
      digitalWrite (lampu1,1);}
  }
  // Jila tidak dipencet
  else {
    digitalWrite (lampu1,0);
  }
}
// Save preference
if (digitalRead (swic) == HIGH){
  tres = LamaPencet;
}
//Sistem sensor
if (sensor == HIGH && koe < 1) {
  digitalWrite (lampu, 1);
  delay (tres-LamaPencet);
  LamaPencet = tres;
  koex = koe+1;
}
if (digitalRead(masuk) == HIGH) {
  koex =0;
}
if (koe == 0) {
  digitalWrite (indikator, 1);
}
else {
  digitalWrite (indikator, 0);
}

condiawal = condi;

```

```

waktuon = mulai1;
waktucet = mulai;
condiawal = condi;
kondisiawal = kondisi;
kondisiawal1= kondisi1;
lamaSebelum = LamaPencet;
koe = koex;
}

```

Kode Untuk Sensor Saja

```

#include <HX711.h>
int beratawal = 0;
int xxx = 5;
int koe = 0;
int koex = 0;
int keluar = 2;
HX711 scale(8,9);
void setup() {
    // put your setup code here, to run once:
    pinMode (keluar, OUTPUT);
    pinMode (xxx, INPUT);
    pinMode (masuk, INPUT);
    Serial.begin(9600);
}

void loop() {
    // put your main code here, to run repeatedly:
    scale.set_gain(128);
    beratawal = scale.read();

    Serial.print ("baca");
    Serial.println (beratawal);
    delay(200);
    Serial.println (digitalRead (keluar));
    if (beratawal < 0) {
        digitalWrite (keluar,1);
    }
    else {

```

```
digitalWrite (keluar,0);
```

```
}  
}
```

Kode Gabungan (Awat menyebabkan pembacaan sistem tombol menjadi kacau)

```
#include <HX711.h>
```

```
int baca = 0;  
int bacax = 0;  
int tombol = 5;  
int tombol1 = 8;  
int LamaPencet = 0;  
int LamaPencet1 = 0;  
int kondisiawal = 0;  
int kondisiawal1 = 0;  
int kondisi= 0;  
int kondisi1= 0;  
int mulai = 0;  
int mulai1 = 0;  
int putus = 0;  
int putus1 = 0;  
int tahan = 0;  
int tahan1 = 0;  
int lamaSebelum = 0;  
int lamaSebelum1 = 0;  
int lepas = 0;  
int lepas1 = 0;  
int swic = 7;  
int lampu = 18;  
int lampu1 = 19;  
int tres = 1000;  
int condi = 0;
```

```

int condiawal = 0;
int waktucet = 0;
int kadal = 0;
int kadall = 0;
int atas = 1500;
int bawah = 0;
int waktuon = 0;
int waktuona = 0;
int inipin = 14;
int inipun = 15;
HX711 scale;
void setup() {
    // put your setup code here, to run once:
    pinMode(tombol, INPUT);
    pinMode(tombol1, INPUT);
    pinMode (swic, INPUT);
    pinMode (lampu, OUTPUT);
    pinMode (lampu1, OUTPUT);
    pinMode (inipin, INPUT);
    Serial.begin(9600);
}

void autokencang()
{
    digitalWrite (lampu1, 1);
    delay (LamaPencet - tres);
    digitalWrite (lampu1, 0);
    LamaPencet = tres;
    Serial.print ("lama");
    Serial.println (LamaPencet);
}

void autokendor()
{
    digitalWrite (lampu, 1);
    delay (tres - LamaPencet);
    digitalWrite (lampu, 0);
    LamaPencet = tres;
    Serial.print ("lama");
    Serial.println (LamaPencet);
}

```

```

}
void loop() {
    // put your main code here, to run repeatedly:
    // Inisiasi kondisi input
    kondisi = digitalRead(tombol);
    kondisi1 = digitalRead(tombol1);
    condi = digitalRead(swic);
    scale.begin (4,15);
    baca = scale.read();

    if (kondisi != kondisiawal) {
// Jika baru dipencet
    if (kondisi == HIGH) {
        mulai = millis();
        if (mulai - waktucet < 700 && waktucet != 0) {
            autokendor();
            LamaPencet = 0;
            bacax = 0;
        }
    }
// Jika baru dilepas
    else {
        putus = millis();
        LamaPencet = lamaSebelum + putus - mulai;
        Serial.print ("lama");
        Serial.println (LamaPencet);
        if (LamaPencet > atas) {
            LamaPencet = atas;
        }
    }
}
// Jika tombol ditahan
else {
    if (kondisi == HIGH){
        tahan = millis();
        kadal = lamaSebelum + tahan - mulai;
        Serial.println (kadal);
        digitalWrite (lampu,1);
        if (kadal > atas){

```



```

    digitalWrite (lampu,0);
    LamaPencet = atas;
}

}
// Jila tidak dipencet
else {
    digitalWrite (lampu,0);
}
}
// TOMBOL KEDUA
if (kondisi1 != kondisiawal1) {
// Jika baru dipencet
if (kondisi1 == HIGH) {
    mulai1 = millis();
    if (mulai1 - waktuon < 700 && waktuon != 0) {
        digitalWrite (lampu1,1);
        delay (LamaPencet);
        LamaPencet = 0;
        waktuon = 0;
    }
    if (kondisi1 == LOW && kondisiawal1 == HIGH) {
        digitalWrite (lampu1,0);
    }
}
}
// Jika baru dilepas
else {
    putus1 = millis();
    LamaPencet = lamaSebelum + mulai1 - putus1;
    Serial.print ("lama");
    Serial.println (LamaPencet);
    if (LamaPencet < 0) {
        LamaPencet = 0;
    }
}
}
// Jika tombol ditahan
else {

```

```

if (kondisi1 == HIGH){
    tahan1 = millis();
    kadall = lamaSebelum + mulai1 - tahan1;
    Serial.println (kadall);
    if (kadall < bawah){
        digitalWrite (lampu1,0);
        LamaPencet = 0;
    }
    else {
        digitalWrite (lampu1,1);}
}
// Jila tidak dipencet
else {
    digitalWrite (lampu1,0);
}
}
// Sistem auto lacing dari tombol auto
condi = digitalRead(swic);
if (condi != condiawal){
    if (condi == HIGH) {
        if (LamaPencet > tres) { //Jika kekencangan lebih kencang dari
        treshold maka dilakukan pengenduran menuju treshold
        autokencang();
        }
        else { //Jika kekencangan lebih kendor dari treshold maka akan
        dilakukan pengencangan menuju treshold
        autokendor();
        }
    }
}
else {
    digitalWrite (lampu,0);
}
}
//Sistem sensor
if (baca < 0 && bacax <1) {
    digitalWrite (lampu,1);
    delay (tres);
    LamaPencet = tres;
    bacax = bacax+1;
}

```

```

}
condiawal = condi;
waktuon = mulai1;
waktucet = mulai;
condiawal = condi;
kondisiawal = kondisi;
kondisiawal1 = kondisi1;
lamaSebelum = LamaPencet;
}

```

Program yang digunakan untuk plotting. Grafik yang digunakan merupakan bacaan dari MegunoLink Pro.

```

#include "MegunoLink.h"
TimePlot MyPlot;

```

```

int baca = 0;
int bacax = 0;
int tombol = 5;
int tombol1 = 7;
int LamaPencet = 0;
int LamaPencet1 = 0;
int kondisiawal = 0;
int kondisiawal1 = 0;
int kondisi = 0;
int kondisi1 = 0;
int mulai = 0;
int mulai1 = 0;
int putus = 0;
int putus1 = 0;
int tahan = 0;
int tahan1 = 0;
int lamaSebelum = 0;
int lamaSebelum1 = 0;
int sensor = 0;
int kenceng = 0;
int kendor = 0;

```

```

int swic = 8;
int lampu = 18;
int lampu1 = 19;
int tres = 200;
int condi = 0;
int condiawal = 0;
int waktucet = 0;
int kadal = 0;
int kadal1 = 0;
int atas = 820;
int bawah = 0;
int waktuon = 0;
int waktuona = 0;
int darisen = 14;
int koe = 0;
int masuk = 4;
int indikator = 9;
int koex = 0;
int tentu = 0;

```

```

void setup() {
    // put your setup code here, to run once:
    pinMode(tombol, INPUT);
    pinMode(tombol1, INPUT);
    pinMode (swic, INPUT);
    pinMode (darisen, INPUT);
    pinMode (lampu, OUTPUT);
    pinMode (lampu1, OUTPUT);
    pinMode (masuk, INPUT);
    pinMode (indikator, OUTPUT);
    Serial.begin(9600);
    MyPlot.SetTitle("Plot waktu tersimpan");
    MyPlot.SetXlabel("Time");
    MyPlot.SetYlabel("Waktu");
    MyPlot.SetSeriesProperties("Lama", Plot::Blue, Plot::Solid, 2,
    Plot::Square);
    MyPlot.SetTitle("Output");
    MyPlot.SetXlabel("Time");

```

```

    MyPlot.SetYlabel("Kondisi");
    MyPlot.SetSeriesProperties("Relay a", Plot::Green, Plot::Solid, 2,
    Plot::Square);
    MyPlot.SetSeriesProperties("Relay a", Plot::Magenta, Plot::Solid,
    2, Plot::Square);

}
void loop() {
    // put your main code here, to run repeatedly:
    // Inisiasi kondisi input
    kondisi = digitalRead(tombol);
    kondisi1 = digitalRead(tombol1);
    condi = digitalRead(swic);
    sensor = digitalRead (darisen);

    if (kondisi != kondisiawal) {
// Jika baru dipencet
        if(kondisi == HIGH) {
            mulai = millis();
            if (mulai - waktucet < 700 && waktucet != 0 && LamaPencet <
tres) {
                kenceng = kenceng +1;
            }
        }
// Jika baru dilepas
        else {
            if (kenceng > 0){
                digitalWrite (lampu,1);
                delay (tres-LamaPencet);
                LamaPencet = tres;
                kenceng = 0;
            }
        }
        else {

    putus = millis();
    LamaPencet = lamaSebelum + putus - mulai;

```

```

        if (LamaPencet > atas) {
            LamaPencet = atas;
        }
    }
}
}
// Jika tombol ditahan
else {
    if (kondisi == HIGH){
        tahan = millis();
        kadal = lamaSebelum + tahan - mulai;

        digitalWrite (lampu,1);
        if (kadal > atas){
            digitalWrite (lampu,0);
            LamaPencet = atas;
        }

    }
    // Jila tidak dipencet
    else {
        digitalWrite (lampu,0);
    }
}
// TOMBOL KEDUA
if (kondisi1 != kondisiawal1) {
// Jika baru dipencet
if (kondisi1 == HIGH) {
    mulai1 = millis();
    if (mulai1 - waktuon < 700 && waktuon != 0 && LamaPencet >
bawah) {
        kendor = kendor +1;
    }
    if (kondisi1 == LOW && kondisiawal1 == HIGH) {
        digitalWrite (lampu1,0);
    }
}
}
}

```

```

// Jika baru dilepas
else {
    if (kendor > 0) {
        digitalWrite (lampu1,1);
        delay (LamaPencet);
        LamaPencet = 0;
        kendor = 0;

    }
    else {

        putus1 = millis();
        LamaPencet = lamaSebelum + mulai1 - putus1;

        if (LamaPencet < 0) {
            LamaPencet = 0;
        }
    }
}

// Jika tombol ditahan
else {
    if (kondisi1 == HIGH) {
        tahan1 = millis();
        kadal = lamaSebelum + mulai1 - tahan1;

        if (kadal < bawah) {
            digitalWrite (lampu1,0);
            LamaPencet = 0;
        }
        else {
            digitalWrite (lampu1,1);}
    }
    // Jila tidak dipencet
    else {
        digitalWrite (lampu1,0);
    }
}

// Save preference

```

```

if (digitalRead (swic) == HIGH){
    tres = LamaPencet;
}
//Sistem sensor
if (sensor == HIGH && koe < 1) {
    digitalWrite (lampu, 1);
    delay (tres-LamaPencet);
    LamaPencet = tres;
    koex = koe+1;
}
if (digitalRead(masuk) == HIGH) {
    koex =0;
}
if (koe == 0) {
    digitalWrite (indikator, 1);
}
else {
    digitalWrite (indikator, 0);
}
if (digitalRead(lampu) == 1) {
    baca = 1000;
}
else {
    baca = 0;
}
if (digitalRead(lampu1) == 1) {
    bacax= 1000;
}
else {
    bacax = 0;
}
if (kondisi || kondisi1 == HIGH){
    tentu = kadal;
}
else {
    tentu = LamaPencet;
}

```



```
MyPlot.SendData (F("Waktu Tersimpan"),tentu);
MyPlot.SendData (F("Kondisi Relay Pengguling"),baca);
MyPlot.SendData (F("Kondisi Relay Pengendur"), bacax);
    condiawal = condi;
    waktuon = mulai1;
    waktucet = mulai;
    condiawal = condi;
    kondisiawal = kondisi;
    kondisiawal1= kondisi1;
    lamaSebelum = LamaPencet;
    koe = koex;

}
```

BIOGRAFI PENULIS



Hibban Kaldera, lahir di Jakarta 27 Agustus 1994. Anak pertama dari dua bersaudara. Penulis memulai jenjang pendidikan di jenjang sekolah dasar di MIN Cijantung (2000-2006). Setelah lulus Sekolah Dasar tahun 2006 penulis melanjutkan ke SMPN 49 Jakarta lalu pindah ke SMPN 2 Mojokerto pada tahun 2007, pada tahun 2009, penulis melanjutkan ke SMAN 1 Sooko Mojokerto setelah lulus pada tahun 2012 penulis melanjutkan studi di Institut Teknologi Sepuluh Nopember jurusan Teknik Elektro dengan konsentrasi di bidang studi Elektronika. Pada masa perkuliahan penulis aktif menjadi anggota berbagai komunitas seperti IST Surabaya dan Trefoil Surabaya serta menjadi founder komunitas Suroboyo Hammers.

email: hibban.monte@gmail.com

